

Amendments to the Claims:

Set forth below in ascending order, with status identifiers, is a complete listing of all claims currently under examination. Changes to any amended claims are indicated by strikethrough and underlining. This listing also reflects any cancellation and/or addition of claims.

1. (currently amended) A graph engine for manipulating data in a database, the graph engine comprising:

a context engine configured to read information from one or more cells derived from standardized database statements as context data blocks, each of the one or more cells including a header and a payload, the header of each of the one or more cells instructing the graph engine how to process the cell;

a read engine configured to read data from the database by matching arguments against entries in the database and returning results from the database; and

a write engine configured to write data into the database by creating an entry in the database and writing data to that entry in the database;

wherein information stored in the database is represented in memory in the form of one or more graph data structures, each graph data structure including one or more sub-trees, wherein the read engine operates by reading data from a location in memory and comparing the contents of the memory location with a search object, the read engine using differential bits between the contents of the memory location and the search object to ~~locate~~retrieve addresses that point to subsequent memory locations in the database, wherein the graph engine is implemented entirely in hardware, and wherein the entire database resides in one of random-access memory and flash memory.

2. (canceled)

3. (previously presented) The graph engine of claim 1, wherein each of the one or more subtrees includes profile data, differential bit matching, and results.

4.-6. (canceled)

7. (previously presented) The graph engine of claim 1, wherein the standardized database statements are Structured Query Language statements.

8. (previously presented) The graph engine of claim 1, wherein the standardized database statements are Extensible Markup Language statements.

9. (previously presented) The graph engine of claim 1 wherein the graph engine is able to process multiple cells representing multiple instructions by pipelining.

10. (withdrawn) A method for manipulating data in hardware database using a graph engine, the graph engine including a context engine, a read engine and a write engine, the method comprising:

passing a search object and a location in a memory containing the database to the context engine; reading the information from a location in memory; comparing the search object and the information using the read engine;

accessing additional locations in memory as a result of the comparison; further comparing the search object to the additional locations in memory; and

returning a result based on the comparisons between the search object and the memory location.

11. (withdrawn) The method of claim 10 wherein the result is a pointer to a new location in memory, the new location in memory to be further compared to a new search object.

12. (withdrawn) The method of claim 10 wherein result is a piece of data stored in the database.

13. (withdrawn) The method of claim 12 further comprising in place of returning a result the step of determining the first differential bit between the search object and the information in memory and writing new information to the database beginning at the first differential bit.

14. (withdrawn) The method of claim 10 wherein manipulating the database is done using standardized database statements.

15. (withdrawn) The method of claim 14 wherein the standardized database statements are Xtensible Markup Language statements.

16. (withdrawn) The method of claim 14 wherein the standardized database statements are Structured Query Language statements.

17. (withdrawn) The method of claim 14 wherein comparing the search object and the information involves comparing differential bits between the search object and the information.

18. (previously presented) A graph engine for manipulating data in a database, the graph engine comprising:

a context engine configured to read information from one or more cells derived from standardized database statements as context data blocks, each of the one or more cells including a header and a payload, the header of each of the one or more cells instructing the graph engine how to process the cell;

a read engine configured to read data from the database by matching arguments against entries in the database and returning results from the database; and

a write engine configured to write data into the database by creating an entry in the database and writing data to that entry in the database;

wherein information stored in the database is represented in memory in the form of one or more graph data structures, each graph data structure including one or more sub-trees, wherein the write engine operates by identifying a first differential bit between the contents of a memory location in the database and a search object, wherein the write engine is configured to create a new entry in the database by writing information beginning at the location of the first differential bit, wherein the graph engine is implemented entirely in hardware, and wherein the entire database resides in one of random-access memory and flash memory.